

Opinnäytetyö (AMK)

Tietotekniikka

Mediatekniikan suuntautumisvaihtoehto

2015

Aapo Peltola

# VIRTUAALITODELLISUUDEN SOVELTAMINEN OPETUS- JA OPASTUSKÄYTÖSSÄ



TURUN AMMATTIKORKEAKOULU  
TURKU UNIVERSITY OF APPLIED SCIENCES

Aapo Peltola

## VIRTUAALITODELLISUUDEN SOVELTAMINEN OPETUS- JA OPASTUSKÄYTÖSSÄ

Työn tarkoituksena oli tutkia virtuaalitodellisuuden hyödyntämistä lasten ja nuorten tieteellisessä opetuksessa interaktiivisten sovellusten avulla. Työ tehtiin Tuorlan Observatoriolle ja sovelluksen aiheeksi valittiin Kuun pinnalla liikkuminen ja Kansainvälisen Avaruusaseman läheisyydessä suoritettava avaruuskävely.

Opinnäytetyön teoriaosuudessa käytiin läpi virtuaalitodellisuusteknologiaa tietokonesovellusten ja pelien piirissä, jonka jälkeen tarkasteltiin käytettävissä olevia pelimoottoreita ja virtuaalitodellisuuslaseja. Näiden pohjalta valittiin opinnäytetyössä tehtävään sovellukseen sopivat työkalut. Sovellus kehitettiin sovitulla tavalla ja se pyritään saamaan käyttöön Tuorlan tiloissa tulevaisuudessa.

Lopputuloksena valmistui virtuaalitodellisuussovellus, joka on tarkoitus ottaa käyttöön Tuorlan Observatoriossa mahdollisesti vuoden 2016 aikana. Sovellus tehtiin käyttäen Unity3D-pelimoottoria ja Oculus Rift DK2 –virtuaalitodellisuuslaseja. Työtä tehdessä hyödynnettiin NASAn arkistoista löytyviä 3D-malleja mahdollisimman tarkkojen ja todellisuutta vastaavien virtuaalimaailmojen luomiseksi.

### ASIASANAT:

digitaaliset pelit, avaruus, tietokoneavusteinen opetus, tietokonepelit, virtuaalitodellisuus, Unity3D

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Information Technology | Media Technology

2015 | 42

Instructor: Principal Lecturer, Ph. D. Mika Luimula

Aapo Peltola

# USAGE OF VIRTUAL REALITY IN TEACHING AND GUIDANCE

The goal of this thesis was to study the usage of virtual reality in applications that can be used to present scientific milestones through interactivity and modern technology. An application would be developed that has two game modes: a walk on the surface of the moon next to an Apollo Lunar Lander and a spacewalk in the lower orbit next to International Space Station. The application was made for Tuorla Observatory.

In the theory part of the thesis virtual reality was researched as a technology in computer applications and gaming. The thesis then compares available hardware and software that can be used to make applications and games using virtual reality headsets and it is decided which ones will be used for this thesis' application.

A virtual reality application was made as a result of this thesis. This application is planned to be in use in Tuorla Observatory facilities possibly in 2016. The application was built using Unity3D game engine and it used Oculus Rift DK2 as the virtual reality headset. NASA archives were used in the making of this thesis to ensure the accuracy of the real-life locations. The archives contain a lot of pictures and 3D models of structures made by NASA in the last 50 years of astronomical study.

## KEYWORDS:

digital games, space, computer-based learning, computer games, virtual reality, Unity3D

# SISÄLTÖ

<b>KÄYTETTY SANASTO</b>	<b>6</b>
<b>1 JOHDANTO</b>	<b>7</b>
<b>2 VIRTUAALITODELLISUUS TEKNOLOGIANA</b>	<b>9</b>
2.1 Nykypäivän mahdollisuudet	10
2.2 Rajoitukset	11
<b>3 VIRTUAALITODELLISUUSLASIT</b>	<b>15</b>
3.1 Vaihtoehtoiset laitteet nyt ja tulevaisuudessa	15
3.2 Oculus Rift DK2	16
<b>4 VIRTUAALIMAILMAN VUOROVAIKUTUS</b>	<b>19</b>
4.1 Ongelmat	19
4.2 Ratkaisut	20
4.3 PAD-ohjaimen käyttö tässä työssä	21
<b>5 PELIMOOTTORIT</b>	<b>22</b>
5.1 CryEngine 3	22
5.2 Unreal Engine 4	22
5.3 Unity 5	23
<b>6 SOVELLUS</b>	<b>25</b>
6.1 Kuukävely	26
6.1.1 Ohjelmointi	27
6.1.2 Grafiikka	28
6.2 Avaruuskävely	31
6.2.1 Ohjelmointi	32
6.2.2 Grafiikka	34
6.3 Äänet	35
6.4 Kahden sovellusmoduulin yhdistäminen	36
6.5 Oculus Rift – tuen lisääminen ja toiminnan varmistaminen	36
6.6 Käyttökokemukset ja jatkokehitys	37
<b>7 LOPUKSI</b>	<b>39</b>

## **KUVAT**

Kuva 1. Havainnointikuva näytön aiheuttamasta liike-epäterävyydestä. (kuva Oculus VR). 13

Kuva 2. Oculus Rift käyttömuotoasetukset (Oculus Configuration Utility v. 0.6.0, Oculus VR 2015) 18

Kuva 3. Esimerkki korkeuskartasta. 28

Kuva 4. Kuukävely kehitysvaiheessa. 30

Kuva 5. Kansainvälinen Avaruusasema pelissä. Taustalla Maapallo. 33

Kuva 6. Kuvankaappaus moninpelistä 38

## **TAULUKOT**

Taulukko 1. Eri virtuaalitodellisuuslaseja vertailussa. 18

## KÄYTETTY SANASTO

6DOF	6 Degrees of Freedom, eli kuuden akselin vapaa liikkuminen
CAVE-järjestelmä	Cave Automatic Virtual Environment, useasta projektorista ja valkokankaasta rakentuva virtuaalitodellisuusympäristö
HMD-laite	Head Mounted Display, päässä pidettävä näyttölaite
Immersio	Virtuaalimaailman uskottavuuden aiheuttama läsnäolon tuntu
VR	Virtuaalitodellisuus, käytännössä mikä tahansa virtuaalimaailma, missä käyttäjä voi liikkua ja toimia. Virtuaalitodellisuustermin käyttö yleistynyt päässä pidettävien virtuaalitodellisuuslaitteiden suosion mukana.

# 1 JOHDANTO

Virtuaalitodellisuuslasit ovat kehittyneet viime vuosina huimasti erityisesti amerikkalaisen Oculus VR –yrityksen ansiosta. Näyttäisikin siltä, että virtuaalitodellisuuslasit ja siihen liittyvät liiketunnistusohjaimet ovat tulossa jäädäkseen. Työn kirjoitusvaiheessa valmista kaupallista tuotetta ei tietokoneille ole vielä saatavissa, mutta kehitysversioita on ollut myynnissä jo muutaman vuoden ajan.

Tämän opinnäytetyön tavoitteena on luoda Tuorlan observatoriolle virtuaalitodellisuuslaseja hyödyntävä interaktiivinen sovellus, jolla voidaan esittää esimerkiksi peruskouluikäisille tieteen suurimpia saavutuksia. Saavutuksista otetaan tähän työhön Apollo-avaruusohjelman mahdollistamat miehitetyt lennot Kuuhun sekä Kansainvälisen Avaruusaseman läheisyydessä tehtävät avaruuskävelyt. Sovelluksen autenttisuuteen panostetaan ja sovelluksessa pyritään käyttämään esimerkiksi NASAn luomia 3D-malleja.

Työn teoriaosuudessa tutkitaan virtuaalitodellisuutta teknologiana, sen mahdollistamia sovelluksia ja sen aiheuttamia rajoituksia ja ongelmia. Vaikka teknologia on kehittynyt 90-luvun laitteista, on siinä vielä paljon asioita, joita pitää ottaa huomioon sekä kehittää pitemmälle. Teknologian tutkimisen lisäksi valitaan työhön sopivat virtuaalitodellisuuslasit eri vaihtoehtoista. Virtuaalitodellisuuslasien pitää olla yhteensopiva tietokoneiden kanssa, sillä suuria ja graafisesti vaativia virtuaalimaailmoja on vaikea toteuttaa mobiililaitteilla. Virtuaalitodellisuusteknologian lisäksi sovellustyön tekemiseen tarvittava pelimoottori valitaan ja valinta perustellaan. Markkinoilta löytyy useita pelimoottoreita, jotka tukevat virtuaalitodellisuuslaseja. Pelimoottoreissa on kuitenkin paljon eroja niin ominaisuuksiltaan kuin hinnoittelultaan.

Teoriaosuuden jälkeen käydään läpi sovelluksen tekemisen eri vaiheet. Virtuaalitodellisuussovellus koostuu kahdesta eri osasta. Ensimmäisessä osassa käyttäjä ohjastaa astronauttia Kuun pinnalla tutkien ympäristöään. Kuun pinnalta löytyy myös Apollo-avaruusohjelmassa käytetty Kuulaskeutuja, jota käyttäjä pääsee lähemmin tutkimaan. Kuulaskeutuja, kuten myös sen ympärillä oleva

Kuun pinta, pyritään luomaan mahdollisimman todenmukaiseksi. Tämän lisäksi käyttäjä voi siirtyä nykyaikaiseen avaruustoimintaan, jossa käyttäjä pääsee lentämään avaruuspuvullaan Kansainvälisen Avaruusaseman ympärille. Molemmissa sovelluksen osissa käyttäjä liikkuu NASAn avaruuspukua mallintavan puvun sisällä luoden uskottavan virtuaalitodellisuuskokemuksen. Todellisuutta jäljittelevän grafiikan lisäksi sovellukseen luodaan syvyyttä ja tunnelmaa luova äänimaailma.

Sovelluksesta on tarkoitus tehdä niin käyttäjäystävällinen, että Tuorlan observatorio voi ottaa sen käyttöön omissa tiloissaan ja hyödyntää sitä erilaisissa tapahtumissa, kuten oppilaitosten luokkaretkillä. Tämän varmistamiseksi sovellusta peluutetaan Turun ammattikorkeakoulun pelilaboratorion tiloissa sekä mahdollisissa messutapahtumissa ennen sovelluksen varsinaista käyttöönottoa.



## 2 VIRTUAALITODELLISUUS TEKNOLOGIANA

Tietokoneissa ja tietotekniikassa hyödynnettäviä kuluttajille hinnoiteltuja virtuaalitodellisuuslaitteita on odotettu jo pitkään. Virtuaalitodellisuusteknologioita on monenlaisia.

Vuonna 1992 Illinoisin Yliopisto Yhdysvalloissa esitteli virtuaalitodellisuusteknologian, joka perustui kolmen tai useamman valkokankaan käyttämisen 1:1-skaalalla olevan virtuaalimaailman piirtämiseen käyttäjän ympärille. Monista hyvistä puolista huolimatta (kuten liikkumisen vapaus ja suurempi katselukulma) tässä teknologiassa on kuitenkin kaksi suurta ongelmaa kotikäyttäjän näkökulmasta. Ensimmäinen ongelma on laitteen vaatima tila. Esimerkiksi Visboxin C4 CAVE (Cave Automatic Virtual Environment) vaatii tilan, jonka leveys on noin 3,6 metriä, pituus 3 metriä ja korkeus 2,3 metriä. Toisena suurena ongelmana on hinta. Edellä mainitun Visboxin CAVE-järjestelmän hintaa ei valmistajan sivuilta suoraan löydy, mutta Visboxin omassa CAVE vs. HMD (Head Mounted Display) -vertailussakin CAVE-järjestelmän huonoksi puoleksi annettiin hinta. Itse laitteiston lisäksi myös hintaa nostaa vaadittava tietokonelaitteisto. HMD-laitteissa on vain yksi näyttö, jonka resoluutio vastaa tietokoneiden työpöytänäyttöjä. CAVE-järjestelmä käyttää useampaa ruutua kuvan näyttämiseen, jolloin tietokoneelta vaadittava laskentateho kasvaa huomattavasti. (Visbox 2015.)

HMD-teknologiaa yritettiin saada koteihin jo 90-luvulla muun muassa japanilaisten pelialan suuryritysten Nintendon ja Segan johdolla. Segan valmistamassa Sega VR -laitteessa oli nykyisistäkin VR-laitteista ja älypuhelimista tuttu liiketunnistusjärjestelmä. Segan kohdalla yritys tosin jäi pelihallien tasolle ja kotikonsoleille tarkoitettua mallia ei koskaan edes julkaistu (The Rev. Rob Times 2015). Nintendo julkaisi omansa, Virtual Boy'n, kesällä 1995, mutta lopetti Virtual Boy'n valmistamisen teknisen vajavaisuuden ja huonojen käyttäjäkokemusten perusteella jo vuonna 1996. (Retro Collect 2015.)

90-luvun yritysten jälkeen virtuaalitodellisuus unohdettiin pitkäksi aikaa. Yksi suurimmista syistä 90-luvun laitteiden ja sovellusten epäonnistumiseen oli tieto-

tekniikan kehityksen vajavaisuus monessa osa-alueessa. Näistä vajavaisuuksista suurimpana ja helpoiten huomattavana ovat virtuaalitodellisuusrilaisissa käytettävät näytöt, itse sovelluksia pyörittävät laitteistot (tietokoneet ja kotikonsolit) sekä liikettä tunnistavat anturit.

Matkapuhelinmarkkinoiden muututtua vuoden 2007 Apple iPhoneen johdolla pienien ja tarkkojen näyttöjen kehitys lähti räjähdysmäiseen kasvuun (Hongkiat 2012). Tämä on mahdollistanut nykyaikaisten VR-laitteiden valmistuksen. Itse VR-laitteiden uuden tulemisen aloitti yhdysvaltalaisen Oculus VR:n valmistama Oculus Rift (Tech Crunch 2014). Oculus Riftin ensimmäinen kehittäjäversio julkaistiin kehittäjämarkkinoille vuonna 2012, ja se on siitä asti nauttinut suurta suosiota median ja kuluttajien keskuudessa. Oculus Riftin saavuttama suosio aiheutti kilpailijoiden ilmestymisen. Vuoden 2015 ja 2016 aikana markkinoille on tulossa useita vastaavanlaisia laitteita. Näistä suurinta huomiota on Oculus Riftin lisäksi saanut elektroniikkajätti Sonyn Playstation VR sekä pelialusta Steamin kehittäjän Valven ja matkapuhelinvalmistaja HTC:n yhteistyössä kehitettävä HTC Vive.

Playstation VR:n suurin ero HTC Viveen ja Oculus Riftiin on sen tukema alusta. Sonyn kehittämänä sen pääalusta on pelikonsoli Playstation 4, eikä vielä ole tarkkaa tietoa Playstation VR:n tuesta muilla alustoilla. HTC Vive ja Oculus Rift on monella tapaa teknisiltä ominaisuuksiltaan yhteneväisiä. Suurimpana erona on Viven mukana tulevat laser-paikantimet, joiden avulla pystytään millimetrin tarkkuudella seuraamaan pelaajan liikkumista maksimissaan 25 m<sup>2</sup>:n alueella. Molempien valmistajien VR-laitteissa tulee olemaan liiketunnistusohjaimet.

## **2.1 Nykypäivän mahdollisuudet**

Vuonna 2012 alkanut uusi aikakausi virtuaalitodellisuuslaitteissa on jo tässä vaiheessa tuottanut paljon erilaisia käyttökohteita niin viihteessä, tieteessä kuin myös opetuksessa. Viihteen ulkopuolella virtuaalitodellisuusrilaseja on hyödynnetty monella tavalla eri teknologioiden kanssa luoden esimerkiksi virtuaalisia

taistelukenttiä, joissa armeijan sotilaslääkäreitä voidaan opettaa hoitamaan haavoittuneita sotilaita keskellä tulitaistelua (Daily Mail 2014).

Virtuaalitodellisuutta voi käyttää myös hieman arkipäiväisemmissä asioissa. Vuoden 2014 kesäkuussa Ranskassa järjestettiin ensimmäinen ensimmäisestä persoonasta kuvattu ja Oculus Rift –laseilla katseltava leikkausoperaatio, jonka tarkoitus oli olla ensimmäinen askel virtuaalitodellisuuden hyödyntämisessä kirurgien koulutuksessa mahdollisimman riskittömässä ympäristössä. (Medical News Today 2014.)

Vaikka yleishyödyllisiä ja viihteen ulkopuolella olevia sovelluksia virtuaalitodellisuudelle on paljon, pidetään silti virtuaalitodellisuuslaseja ennen kaikkea peliteollisuuden uutena suuntana. Pelit ovat siinä mielessä looginen alusta virtuaalitodellisuudelle, että suurin osa tarvittavasta muusta teknologiasta on jo olemassa. Tämän lisäksi pelien teknologioiden kehittyessä pelimaailmat alkavat lähestyä ulkonäöllään ja interaktiivisuudellaan tosimaailmaa. Tämän immersion kuitenkin rikkoo näyttöpäätteiden käyttäminen kuvan vastaanottamisessa ja monella tapaa myös pelihahmojen liikuttaminen pelimaailmassa. Virtuaalitodellisuuslasit ja niiden mukanaan tuomat uudet ohjaimet ja ohjaamistavat (ks. luku 4.2), pyrkivät tulevana vuosina muuttamaan tämän.

## 2.2 Rajoitukset

Vaikka teknologia on kehittynyt viime vuosina huimaa tahtia virtuaalitodellisuuslaitteissa ja uusia kilpailijoita ilmoittautuu markkinoille jatkuvasti, on teknologia kuitenkin vielä monilla tavoin rajoittunut. Vaikka suurimmassa osassa muussa tietotekniikassa näyttötarkkuuden standardiksi on muodostunut 1920 kuvapistettä (leveys) kertaa 1080 kuvapistettä (korkeus), on se virtuaalitodellisuuslaseissa auttamatta liian pieni. Kyseinen resoluutio riittää esimerkiksi tietokoneen näytöllä, televisiossa tai älypuhelimessa siihen, että ihmissilmä ei normaaliikäyttöetäisyyksiltä pysty tunnistamaan yksittäisiä kuvapisteitä tai ei näe niitä häiritsevästi. Virtuaalitodellisuuslaseissa taas näyttö peittää käyttäjän lähes koko näkökentän, jolloin näytön tarkkuuden pitäisi olla huomattavasti korkeampi,

kuin mitä ne tämänhetkissä laitteissa on. Tätä ongelmaa on auttamassa koko ajan kehittyvät älypuhelimet, jotka käyttävät vastaavanlaisia ja –kokoisia näyttöjä.

Näyttöjen tarkkuuden lisäksi virtuaalitodellisuudessa on ongelmana pelaajan vuorovaikutteisuus pelin tai sovelluksen kanssa. Pelikonsoleilla tutuksi tulleet ohjaimet tai tietokoneella käytetty näppäimistö ja hiiri eivät enää välttämättä riitä, sillä varsinkin näppäimistön kanssa tulee ongelmaksi pelaajan todellisen maailman näkemisen menettäminen.

Vuorovaikutteisuuteen liittyy myös toinenkin ongelma, simulaatiopahoinvointi. Kyseessä on matkapahoinvointia sivuava terveydentila, joka johtuu tasapaino- aistin ja silmien antaman liikkeen ristiriidasta (Arns & Cerney, 2005). Virtuaalitodellisuuslaseilla ongelma saattaa tulla esimerkiksi siitä, että pelissä tapahtuva liike ei vastaa pelaajan todellisen maailman liikkeitä. Virtuaalitodellisuuslasit jäljittävät pelaajan pään liikkeitä ja muuntavat ne peliin vastaamaan alkuperäistä liikettä ja täten kääntää ja siirtää pelin kameraa ja pelaajan näkökenttää. Pelaajan pään liikkumisen lisäksi on kuitenkin pelaajaa liikutettava myös muullakin tavalla. Yleensä tämä hoidetaan tavalla, jossa virtuaalimaailman liike ja todellisen maailman liike eivät vastaa toisiaan. Tämä aiheuttaa liikkeitä pelin kamerassa, mitä pelaaja ei välttämättä täysin pysty ennakoimaan (vrt. auton takapenkillä istuminen).

Simulaatiopahoinvoinnin korjaamiseksi on kehitteillä monia eri ratkaisuja. Yksinkertaisimmat ovat kädessä pidettävät liiketunnistusohjaimet, joiden liike pyritään toistamaan pelissä ja täten parantaa immersiota ja pelin tapahtumien ennakointia (Sixense Entertainment 2015). Monimutkaisempien ratkaisujen päässä on esimerkiksi Virtuix Omni, jossa pelaaja seisoo pyöreällä juoksumatolla (Virtuix 2015). Pelaajan kävely ja juoksu toistetaan peliin hahmon liikkeeksi antaen sille oikean nopeuden ja suunnan.

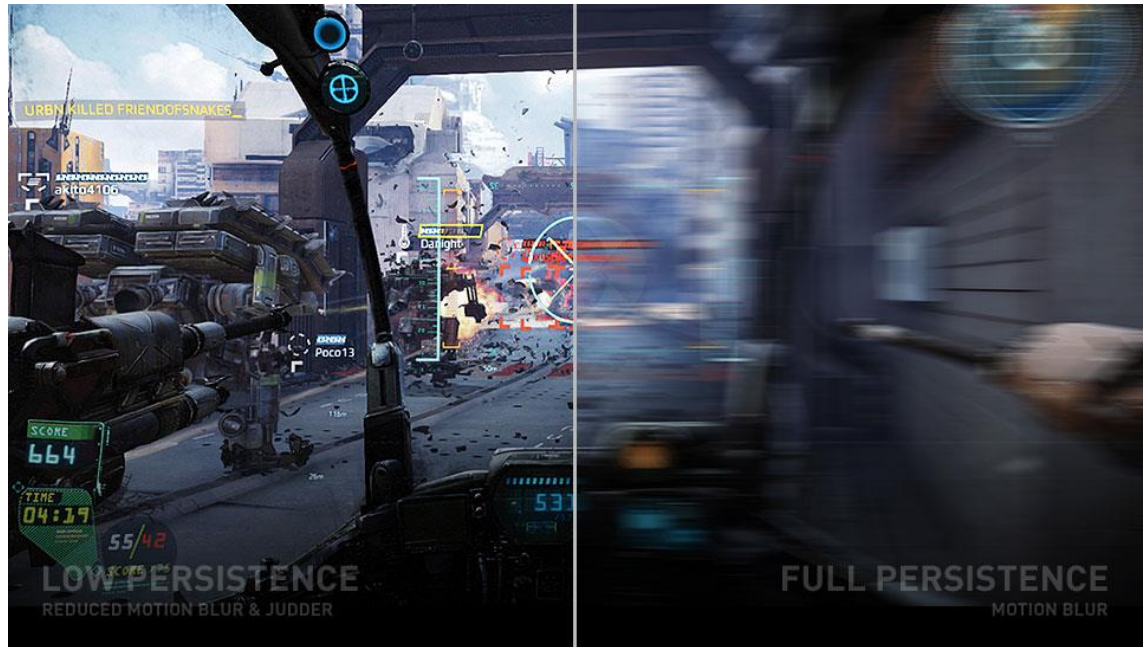
Näiden lisäksi myös pelaajan liikettä voidaan seurata kokonaisuudessaan. Microsoft julkaisi jo vuonna 2010 Xbox 360 –pelikonsolille tarkoitetun Kinect-liiketunnistuskameran. Se sijoitetaan television päälle ja siinä hyödynnetään

normaalin kameran lisäksi infrapunakameraa syvyyden mittaamiseen (Derek Hoiem 2011). Microsoftin Kinectin lisäksi HTC julkisti vuoden 2015 GDC-messuilla Valven kanssa yhteistyössä kehitettävän HTC Vive virtuaalitodellisuuskokonaisuuden, jossa pelaajan liikkeitä voidaan seurata kädessä pidettävien liiketunnistusohjaimien lisäksi koko huoneen kattavalla laserpaikantimella (HTC 2015). Tämän avulla pelaaja voi vapaasti liikkua 25 m<sup>2</sup>:n alueella ja saada reaaliajassa omat liikkeensä peliin, jolloin suurin osa pelaajan vuorovaikutuksen aiheuttamasta simulaatiopahoinvoinnista saadaan poistettua.

Käyttäjän liikkumisen lisäksi yksi vaikuttava tekijä sekä immersion parantamiseen, että simulaatiopahoinvoinnin vähentämiseen on virtuaalitodellisuuslaseissa käytetty näkökentän suuruus. (Fleming Seay, Krum, Hodges & Ribarsky 2001) Näkökentän suuruus vaikuttaa siihen, kuinka paljon pelissä tai muussa sovelluksessa käytettävä kamera näyttää keskipisteeseen nähden. Yleensä tämä määrä ilmaistaan asteissa. Peleissä käytetään yleensä 50 - 110 asteen horisontaalista näkökenttää, verrattuna ihmisen noin 180 – 200 asteen näkökenttään (Environmental Resources Management Australia 2009), riippuen pelityypistä ja käytössä olevasta laitteistosta. Joillain ihmisillä simulaatiopahoinvoinnin oireita saattaa ilmetä jo normaalilla näytöllä läheltä pelatessa, josta johtuen tietokonepeleissä käytetään useasti konsolipelejä suurempia näkökenttiä. Virtuaalitodellisuuslaseissa tämä kuitenkin tarkoittaa sitä, että suuremmalla näkökentällä varustetut lasit peittävät suuremman osan näkökentästä, jolloin lasien näyttämä virtuaalimaailma vastaa lähemmin todellista maailmaa.

Vuorovaikuttamisen lisäksi virtuaalitodellisuuslaseissa on yksi tekninen ongelma mikä olisi hyvä saada lähes kokonaan pois kaupallisten versioiden julkaisuun mennessä. Tämä ongelma on pelaajan pään liikkeen rekisteröimisen, sen lähettämisen pelille, rekisteröidyn liikkeen muuttaminen pelissä tapahtuvaksi liikkeeksi ja uuden kuvan lähettäminen pelaajalle mahdollisimman pienessä ajassa. Monet valmistajat ovat ottaneet minimivaatimukseksi 90:en kuvan sekuntipäivityksen, joka tarkoittaa sitä, että peli joutuu piirtämään pelin tapahtumia ruudulle yhdeksänkymmentä kertaa sekunnissa (Oculus 2015). Ajassa tämä taas tarkoittaa sitä, että uusi kuva täytyy piirtää ruudulle noin 11 ms:n välein.

Ongelma ilmenee siinä, että kuvien vaihtuessa nopeasti ja antureiden syöttäessä koko ajan pelin kameralle liikedataa, näytössä näkyvä kuva saattaa jo olla vanhaa tietoa, sillä ruudulla näkyvä kuva vaihtuu vain 11 ms:n välein (Kuva 1.)



Kuva 1. Havainnointikuva näytön aiheuttamasta liike-epäterävyydestä. (Oculus VR 2015)

Oculus VR on omassa laitteessaan pyrkinyt minimoimaan tämän kuvan ”peräsäläahaamisen” muuttamalla näytön toimintaperiaatetta. Sen sijaan, että piirretty kuva pidetään näytöllä niin kauan ennen kuin tietokone lähettää uuden kuvan piirrettäväksi, kuva näytetään katsojalle vain hetken. Tämän jälkeen näytön kuvapisteen sammutetaan. Toimenpide ei aiheuta kuvan vilkkumista, koska kuvia piirretään niin nopeasti ruudulle. Se kuitenkin poistaa pelaajalle näytettävästä kuvasta liike-epäterävyyttä ja täten vähentää simulaatiopahoinvointia. (Road to VR 2015)

### 3 VIRTUAALITODELLISUUSLASIT

Vuoden 2015 syksyllä ei vielä kuluttajaversioita ole mistään uusimman sukupolven tietokoneille suunnatuista virtuaalitodellisuuslaseista. Ainoa kuluttajille tarkoitettu laite on Samsungin ja Oculus VR:n yhteistyössä kehittämä virtuaalitodellisuusalusta, Samsung Gear VR, jossa näyttönä ja tehonlähteenä toimii Samsungin uudet puhelimet. Tämä kokonaisuus on kuitenkin tarkoitettu käyttämään Android-pohjaisia sovelluksia. Samsung Gear VR:n ongelma on myös hinta, sillä tarvittava mobiililaite maksaa huomattavasti enemmän kuin esimerkiksi Oculus Rift DK2. Kehityksen alla on useita laitteita, kuten HTC Vive ja Sony Playstation VR.

#### 3.1 Vaihtoehtoiset laitteet nyt ja tulevaisuudessa

Työn lopussa kehitettävä sovellus on kuitenkin tarkoitus ottaa käyttöön vasta siinä vaiheessa, kun loppukäyttäjille tarkoitettuja laitteita on saatavilla. Työn tekovaiheessa tämä tarkoitti käytännössä loppuvuotta 2015 tai kevättä 2016. Tästä johtuen on kuitenkin hyvä selvittää mahdollisia laitteita, joita lopputuotteessa voitaisiin käyttää. Sovellus voitaisiin myös mahdollisesti muokata käytettäväksi Android-pohjaisella laitteella, jolloin käytössä olisi esimerkiksi Googlen kehittämä Cardboard –lisävaruste, jossa Android-käyttöjärjestelmää käyttävä puheli toimisi näyttönä ja korvaisi tietokoneen tehonlähteenä tai aikasemmin mainittu Samsung Gear VR mikäli laitteiston kokonaishinta hieman laskee. Pelimoottorina käytetty Unity tukee useampaa alustaa ja eri versioiden tekeminen olisi helppoa. Tämä tosin tarkoittaisi sitä, että sovelluksessa käytettävä ohjaustapa pitäisi mahdollisesti muuttaa Androidia tukevaksi.

Tällä hetkellä markkinoilla ja kehitteillä olevia laitteita on erittäin vaikea verrata toisiinsa. Vain Samsungin Gear VR on lopullinen laite, jonka tekniset tiedot ovat varmoja. Muut laitteet ovat vielä vaiheessa, jossa lopullisia tietoja ei vielä ole saatavilla ja tällä hetkellä tiedossa olevat tekniset yksityiskohdat saattavat

vielä muuttua. Yksi suurimmista selkeästi eroa tekevistä asioista on laseissa käytetyn näytön tarkkuus. Monissa peleissä standardiksi muodostunut HD1080-resoluutio, jossa näytön leveys on 1920 kuvapistettä ja korkeus 1080 kuvapistettä, ei enää riitä virtuaalitodellisuuskaskeissa, sillä kuvaa tarkkaillaan erittäin läheltä. Tämän lisäksi kolmiulotteisen stereokuvan saamiseksi näyttö jaetaan kahteen osaan molemmille silmille, joten yhdelle silmälle tämä tarkoittaa puolta täyden näytön resoluutiosta. Käytännössä siis HD1080-resoluutiossa yhdelle silmälle jää nähtäväksi vain 960 kuvapistettä leveydessä.

Taulukko 1. Eri virtuaalitodellisuuskaskeja vertailussa.

LAITE	Näytön tarkkuus silmää kohden	Tuettavat alustat	Julkaisu
Samsung Gear VR	1280 x 1440	Android	2014
Oculus Rift DK2	960 x 1080	PC, Mac, Linux	2014, kaupallinen versio kevät 2016
HTC Vive Dev Kit	1200 x 1080	PC, Mac, Linux	2014, kaupallinen versio syksy 2015
Sony PlayStation VR	960 x 1080	PS4, PC*	Ei saatavilla, kaupallinen versio 2016

\* Lopulliset yksityiskohdat vielä varmistamatta

Sovelluksen käyttöönottoajankohta on siis aikaisintaan keväällä 2016, jolloin markkinoilla pitäisi olla saatavilla jo useampi laite ja paremman vertailun voi tehdä vasta silloin.

### 3.2 Oculus Rift DK2

Tämän työn toteuttamiseen valittiin Oculus Rift DK2 lähinnä saatavuuden takia. Työn loppuvaiheessa HTC oli jo alkanut jakaa kehittäjäversioita Viveä, mutta sellaisen käyttöön ei tätä tehdessä ollut mahdollisuutta. Samaten Samsung

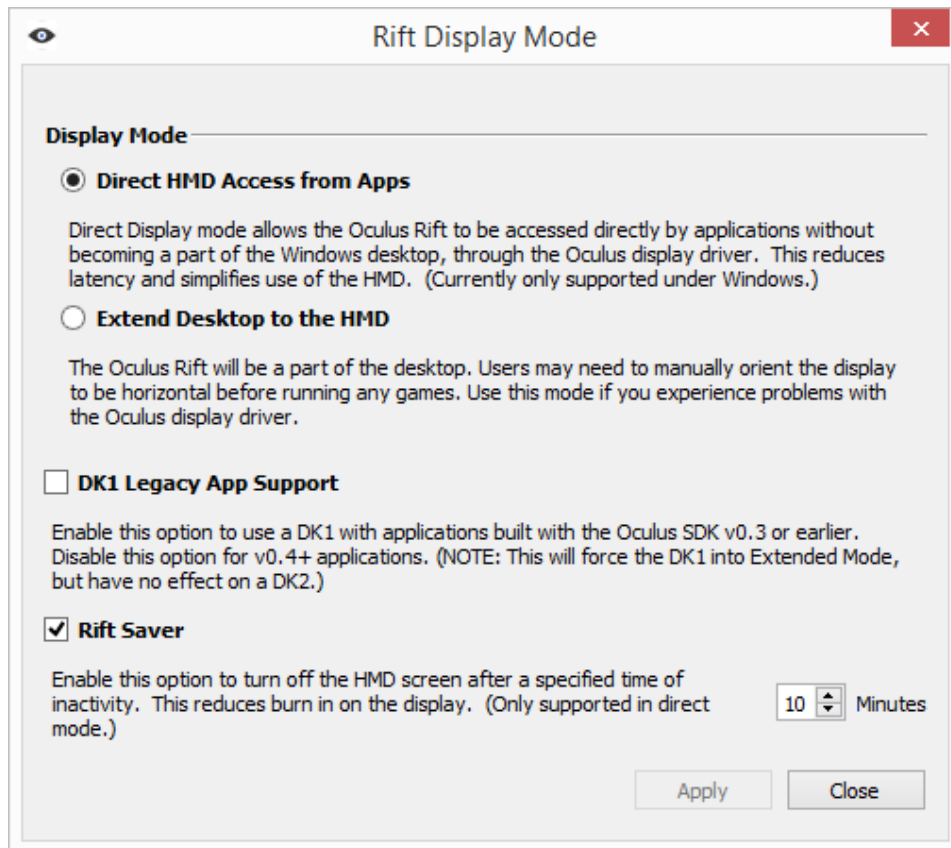


Gear VR ja Google Cardboard jätettiin pois suurimmaksi osaksi mobiililaitteiden tehonpuutteen takia.

Oculus Rift DK2 on Oculus VR:n kehittämän virtuaalitodellisuuslasien kehittäjä-laitteen toinen versio. Ensimmäinen versio julkaistiin vuonna 2012, joka monine ongelmineenkin osoitti monelle 2010-luvun virtuaalitodellisuuslasien potentiaalin. DK2 on kehittyneempi versio ensimmäisestä monellakin osa-alueella. Näyttö on tarkempi ja nopeampi, pään kääntymisen tunnistavat anturit toimivat tehokkaammin ja ensimmäisestä versiosta poiketen DK2 myös sisältää käyttäjän eteen asennettavan kameran ja laitteessa itsessään sijaitsevat infrapuna-LEDit, joiden avulla voidaan kääntymisen lisäksi myös tulkita käyttäjän pään sijainti. Monet näistä uusista ominaisuuksista oli suunniteltu parantaa käyttäjämukavuutta. Tarkempi ja nopeammin reagoiva näyttö sekä pään liiketunnistus taas immersion parantamisen lisäksi myös vähentävät lasien aiheuttamaa simulatiopahoinvointia.

Oculus Rift DK2:sta löytyy kaksi eri käyttömuotoa. Ensimmäinen ja vanhempi muoto on jo Oculus Rift DK1:ssä käytössä ollut muoto, missä Oculus Riftin näyttö käyttäytyy normaalina näyttönä ja täten on Windowsin työpöydän jatkona. Tämä on yksinkertaisuutensa ansiosta varmatoimisempi ja yleisemmin tuettu, mutta aiheuttaa suuremman vasteajan kuvan ja liikkeen välillä. Oculus VR suosittelee tätä käytettäväksi vain silloin, kun Oculus VR:n uudempi muoto, Direct HMD Access, ei toimi.

Direct HMD Access toimii hieman eri tavalla ensimmäiseen käyttömuotoon nähden. Siinä ei Oculus Riftin näyttöä käytetä työpöydän jatkeena missään vaiheessa, vaan kuva lähetetään suoraan pelistä näytönohjaimen kautta Oculus Riftin ruudulle (Kuva 2). Tässä on hyötynä esimerkiksi se, että käyttöjärjestelmä jää välistä kokonaan pois. Tämä taas pienentää vasteaikaa kuvan ja liikkeen välillä. Tämä käyttömuoto on se, josta Oculus VR pyrkii saamaan standardin peleihin.



Kuva 2. Oculus Rift käyttömuotoasetukset (Oculus Configuration Utility v. 0.6.0, Oculus VR 2015)

## 4 VIRTUAALIMAILMAN VUOROVAIKUTUS

Siirryttäessä perinteisistä videopeleistä virtuaalitodellisuuslaseja hyödyntäviin hyöty- ja viihdesovelluksiin yksi suurimmista kehittämiskohteista on käyttäjän vuorovaikutus virtuaalimaailman kanssa. Virtuaalilasien luoma immersio virtuaalimaailmassa olemisesta korostaa vuorovaikutuksen tärkeyttä. Jos pelaaja tuntee olevansa virtuaalimaailmassa katsellessaan ympärilleen, mutta joutuu tämän jälkeen liikkumaan käyttäen perinteisesti esimerkiksi näppäimistöä ja hiirtä, saattaa se rikkoa virtuaalitodellisuuslasien aiheuttaman immersion täysin.

### 4.1 Ongelmat

Teknisesti ajateltuna yksi suurimmista ongelmista virtuaalitodellisuuslaseja käyttäessä vuorovaikutuksessa ja pelin ohjaamisessa on käyttäjän oikean maailman näkemisen puuttuminen. Tämä aiheuttaa sen, että käyttäjä ei voi pelin aikana esimerkiksi helposti vaihtaa käden paikkaa ohjainlaitteella. Tämä aiheuttaa ongelmia erityisesti näppäimistöä käyttäessä. Käyttäjän on vaikea painaa haluttuja painikkeita näppäimistöllä, koska ei näe käsiensä sijaintia näppäimistöllä.

Toisena ongelmana on perinteisten ohjaustapojen staattisuus. Käyttäjä istuu paikallaan ja antaa sormillaan komentoja näppäimistöllä, hiirellä tai vaikka peliohjaimella. Tämä taas aiheuttaa pelissä hahmon liikkumista, hyppimistä ja muita liikkeitä, jotka eivät kuitenkaan vastaa käyttäjän tekemiä liikkeitä oikeassa maailmassa. Tämä aiheuttaa jo työn aikaisemmassa vaiheessa puhuttua simulaatiopahoinvointia. Pahoinvointi ja huimaus tulevat siitä, että käyttäjän tasapainoasti väittää käyttäjän olevan täysin paikoillaan, kun taas silmät lähettävät signaalia aivoille koko ajan tapahtuvasta liikkeestä virtuaalimaailmassa.

Kolmantena kohtana vuorovaikutuksen ongelmassa on immersion rikkoutuminen. Nykypäivänä pelien graafinen ulosanti on kehittynyt niin pitkälle, että oikeanlaisilla laitteilla pelimaailmaan pystyy uppoutumaan erittäin tehokkaasti. Virtuaalitodellisuuslasit vahvistavat tätä virtuaalimaailmaan uppoutumista niin te-

hokkaasti, että immersion rikkoutumisen huomaa perinteisiä pelitapoja helpommin. Immersion rikkoutuminen tapahtuu helposti silloin, kun pelaajan oikeassa maailmassa tekemät asiat eivät vastaa pelissä tapahtunutta. Esimerkkinä voisi pitää vaikka peliä, jossa pelaaja liikkuu virtuaalisessa huoneessa, mutta todellisessa maailmassa pelaaja istuu vain paikallaan ja käyttää peliohjainta hahmon liikuttamiseen.

## 4.2 Ratkaisut

Virtuaalitodellisuuslaseja kehittävät yritykset yrittävät lasien lisäksi myös kehittää uusia tapoja pelien vuorovaikutukseen. Lähimpänä perinteisiä ohjaimia on käsissä pidettävät liiketunnistusohjaimet.

Parhaimpana esimerkkinä liiketunnistusohjaimesta on yhdysvaltalaisen Sixense Entertainment, Inc.:n valmistama STEM System, jossa käyttäjällä on käsissään ohjaimet, jotka tunnistavat kaikki ohjaimen liikkeet ja asennot. Tämän lisäksi pelaajalla on käytettävissä useita painikkeita ja monisuuntainen ohjaussauva. Näiden lisäksi STEM System sisältää eri ruumiinosiin kiinnitettäviä liikeantureita, joiden avulla pystytään tunnistamaan käyttäjän kaikki liikkeet. Kokonaisuudessaan laitteisto tukee järjestelmää, missä liiketunnistimia on käyttäjän jaloissa ja keskiruumiissa, jolloin käytännössä pelaaja voi kävellä ympäri huonetta niin, että käyttäjän kaikki liikkeet voidaan toistaa pelissä. (Sixense Entertainment 2015)

Fyysisten liiketunnistimien lisäksi on olemassa muunlaisia järjestelmiä liikkeiden tunnistamiseen. Microsoft on julkaissut jo kaksi versiota omasta Kinect-järjestelmästä missä hyödynnetään videokameraa ja infrapunakameraa käyttäjien liikkeiden seuraamiseen. Uusin kilpailija tämäntapaisissa laitteissa on Valve Corporationin ja HTC:n yhteisprojekti HTC Vive, jossa virtuaalitodellisuuslasien ja käsissä pidettäviä liiketunnistusohjaimien lisäksi käyttäjä voi ottaa käyttöönsä lasereita hyödyntävän järjestelmän, joka seuraa pelaajan liikettä ja voi sen toistaa peliin. Tämä järjestelmä tukee alustavien tietojen mukaan kahdenkym-

menenviiden neliömetrin tilaa. Tämän avulla pelaajan liikkeet pyritään toistamaan pelissä niin tarkasti, että pelaaja unohtaa olevansa virtuaalimaailmassa.

Näiden järjestelmien lisäksi tietynlaisissa peleissä voidaan hyödyntää suoraan perinteisiä ohjaintapoja. Lentosimulaattorien pelaaminen ohjaussauvan ja lentokoneista tutun kaasukahvan avulla luo erittäin todenmukaisen kokemuksen lentokoneella lentämisestä ilman g-voimien aiheuttamaa räsytystä. Samalla tavalla ratti-ohjaimilla pystyy erittäin tarkasti simuloimaan kilpa-autolla ajamista virtuaaliodellisuuslaseilla tehostettuna. Näissä molemmissa tapauksissa hyvänä puoleena on myös se, että pelaajan hahmo pysyy lähiympäristöön nähden paikallaan, joten pelaajalla on koko ajan lähellä kiintopiste, mikä ei liiku pelaajan sijaintiin nähden.

#### **4.3 PAD-ohjaimen käyttö tässä työssä**

Tämän työn käytännön sovellusta suunnitellessa yksi tärkeimmistä asioista oli käyttäjän vuorovaikutuksen yksinkertaisuus. Pelihahmon ohjaaminen pitää olla helposti opittavissa, koska käyttäjillä ei ole käytettävissä paljoa aikaa. Vuorovaikutuksen piti myös tapahtua sellaisella laitteistolla, jonka hankkiminen olisi taloudellisesti mahdollista.

Näiden lähtökohtien avulla päädyttiin käyttämään Microsoftin XBOX 360 –pelikonsolin ohjainta. Kyseisessä peliohjaimessa on kaksi pientä analogista sauvaohjainta peukaloille, kaksi analogista liipaisinpainiketta etusormille sekä usea digitaalinen painike. Ohjaimen käyttö on erittäin helppoa ja ohjaimen muotoilun ansiosta se soveltuu monien käsiin. XBOX 360 –peliohjain on myös erittäin edullinen valinta ja siitä on olemassa sekä langallinen että langaton versio. Näiden kahden version välillä voidaan tehdä päätös, kun sovellus otetaan käyttöön.

## 5 PELIMOOTTORIT

Jotta voitaisiin luoda uskottavia virtuaalimaailmoja tutkittavaksi, on se luotava alustalle, jolla pystyy rakentamaan tarvittavan vuorovaikutteisuuden sekä luomaan graafinen ulkoasu, joka vastaa tarpeita. Tätä alustaa kutsutaan pelimoottoriksi, joka pitää sisällään kaiken tarvittavan fysiikan mallinnuksesta äänimaailman luontiin. Pelimoottorit koostuvat useista paloista, jotka yksinkertaisimmillaan ovat ohjelmoinnin tuoma pelilogiikka, grafiikan hahmontaminen, fysiikan mallintaminen, äänen luominen ja tekoäly (Game Career Guide 2008).

Tähän työhön valittiin tutkittavaksi kolme pelimoottoria, jotka sopivat ominaisuuksiltaan tähän työhön. Aikaisempaa kokemusta kyseisistä pelimoottoreista oli vaihtelevasti, jotka myös vaikuttivat päätöksen tekemiseen.

### 5.1 CryEngine 3

CryEngine 3 on saksalaisen peliyrityksen, Crytekin, alun perin sisäiseen käyttöön tehdyn pelimoottorin viimeisin versio, joka on nykyään yksi varteenotettavimmista lisensoitavista pelimoottoreista. CryEngine 3 on yksi graafisesti näyttävimmistä pelimoottoreista, mutta sitä pidetään vaikeampana oppia, eikä välttämättä ole paras vaihtoehto, ellei tehtävän pelin ole tarkoitus olla graafisesti erittäin vaativa (Digital Tutors 2015).

Uusimpien versioiden myötä CryEngine tukee myös virtuaalitodellisuuslaitteita, joka mahdollistaisi sen käytön myös tässäkin työssä.

### 5.2 Unreal Engine 4

Tämän kolmikon ehkä historiallisesti merkittävin ja pisimpään olemassa ollut pelimoottori on Unreal Engine. Unreal Engine sai alkunsa 90-luvun lopulla ja näki päivänvalon Epic Gamesin vuoden 1998 toimintapelissä Unreal. CryEnginestä poiketen Unreal Engineä lisensoitiin muille peliyrityksille alusta alkaen.

Viimeistään Unreal Engine 3:n myötä siitä tuli pelialan suosituin lisensoitava pelimoottori suurien AAA-pelien pohjana.

Unreal Enginen suosio perustuu sen pitkälle kehittyneisiin työkaluihin ja tehokkaaseen graafiseen ulosantiin. Uusimman version myötä suosio on kasvamassa myös pienempien ja aloittelevien yritysten parissa, sillä aikaisemmin käytössä ollut kertalisenssimaksu on poistunut ja nykyisin pelimoottorin saa täysin käyttöönsä maksutta. Kertalisenssimaksun tilalle tuli pelin tuottojen mukaan menevät rojaltimaksut. Nykyisin myös kuka tahansa Unreal Engineä käyttävä pelinkehittäjä saa pelimoottorin lähdekoodin ja täten voi muokata pelimoottorin vastaamaan täysin omia tarpeita.

Työn alkuvaiheessa tarkoituksena oli tehdä sovellus nimenomaan Unreal Enginellä hyödyntäen kolmannen osapuolen kehittämää C# -ohjelmointikielen liitännäistä. Liitännäisen asentaminen onnistui toivotulla tavalla, mutta käytännönkokeilun jälkeen päädyttiin vaihtamaan pelimoottoria muutamastakin syystä. Ensimmäisenä syynä oli tiukka kehitysaikataulu, johon ei sittenkään soveltunut kokonaan uuden kehitysympäristön opetteleminen. Aikaisempaa kokemusta Unreal Enginestä oli vain vanhemmista versioista ja C# -ohjelmointikielellä toteutettu kehitysympäristö oli täysin tuntematon. Tämän lisäksi ongelmia jo alkuvaiheessa aiheutti liitännäisen huono dokumentointi ja tekninen tuki. Liitännäisen kehittänyt Xamarin ei tarjoa teknistä tukea, sillä se on tehty lähinnä kokeilumielessä ja yrityksen sisäisenä insinööriprojektina (Mono For Unreal 2015). Liitännäisen käytössä oli myös taloudellinen puoli, sillä Xamarinin valmistamalla liitännäisellä julkaistut tuotteet vaativat Xamarinin lisenssin.

### 5.3 Unity 5

Unity3D-pelimoottori on alunperin tanskalaisen kolmikon kehittämä pelimoottori, jonka pääasiakkaat ovat alusta asti olleet pienet ja yksityiset peliyritykset, joilla ei ole varaa kehittää omia pelimoottoreita. Pitkään Unity3D oli nimensä mukaisesti vain 3D-pelimoottori ja tarkoitettu pienehköihin peleihin, joiden suurimpana myyntivalttina ei ollut ulkonäkö. Kuitenkin loppuvuodesta 2013 Unity päivittyi

versioon 4.3, joka toi mukanaan uudet 2D-työkalut, jotka mahdollistivat monipuolisemmin ja helpommin luotavat kaksiulotteiset pelimaailmat.

Vuonna 2015 Unity Technologies julkaisi pelimoottoristaan version 5.0, jonka tarkoituksena on saada pelimoottorista suurempi kilpailija markkinoilla olevien isompien pelimoottorien, CryEngine 3:n ja Unreal Engine 4:n, kanssa. Se toi mukanaan muun muassa monia graafisia uudistuksia. Unity 5 jatkaa edeltäjiensä siinäkin mielessä, että pelimoottori tukee 21 eri laitealustaa, jolla pyritään nostamaan pelimoottorin kiinnostavuutta (Unity 2015).

Unity-pelimoottori tukee natiivisti Oculus Riftiä ja tulevaisuudessa todennäköisesti myös muita virtuaalitodellisuuslaseja ja -laitteita. Aikaisempien kokemusten perusteella Oculus Riftin saaminen toimimaan Unitylla ei vaadi suurtakaan työtä, vaan peruskäyttöön sen saa Unity Technologiesin valmiiksi tehdyillä työkaluilla.

Tähän projektiin valittiin pelimoottoriksi Unity lähinnä sen helppokäyttöisyyden ja aikaisemman kehityskokemuksen takia. Unity-pelimoottorilla on erittäin nopea kehittää yksinkertaisia prototyyppejä ja vähän monimutkaisempiakin kokonaisuuksia, joka sopi tämän projektin aikatauluun.



## 6 SOVELLUS

Sovelluksen päätarkoituksena oli kehittää virtuaalitodellisuusraseja hyödyntävä hyöty- ja opetussovellus Tuorlan observatorion käyttöön. Sovelluksen suunnittelu alkoi jo ennen Tuorlan liittymistä projektiin, mutta jo varhaisessa vaiheessa sovelluksen suunnitteluun osallistui myös Tuorlan observatorion tutkijatohtori Pasi Nurmi. Suunnitelmissa päädyttiin kahteen erilaiseen ja historiallisesti merkittävään tapahtumaan: Kuu-kävely Apollo-kuumoduulin lähellä sekä avaruus-kävely Kansainvälisen Avaruusaseman ympärillä. Tapahtumat pyritään toistamaan mahdollisimman tarkasti, mutta kuitenkin niin, että sovelluksessa täyden realistisuuden edelle menisi käyttömukavuus, viihdyttävyys ja informatiivisuus. Tällä on sovelluksen sisältöön sellainen vaikutus, että taivaankappaleiden suhteellinen koko ei välttämättä vastaa oikean maailman vastineita, jos kokoa muuttamalla kokemuksesta on saatu jollain tapaa parempi.

Työn teoria-osuudessa käytiin läpi teknologioita, joita sovellusta tehdessä käytettiin. Pelimoottoriksi valittiin Unity 5 suurilta osin sovelluksen tekijän kehittäjätaustan perusteella. Toisena vartenotettavana vaihtoehtona oli Unreal Engine 4, jota työn aivan alkuvaiheessa käytettiin. Alkuvaiheessa kuitenkin kävi ilmi seikkoja, jotka muuttivat suunnitelmaa niin, että Unreal Engine 4 vaihdettiin Unity 5:een. Suurimmat syyt olivat Unreal Engine 4:n C#-tuen lisensointimaksut sekä suuresti aikaa vaativa uusien työkalujen opettelu.

Laitteistopuolella virtuaalitodellisuuslasien suhteen oli käytännössä vain yksi vaihtoehto, Oculus Rift DK2. Työtä tehdessä Oculus Rift oli yksi harvoista laitteista, joita oli kaupallisesti saatavilla. Virtuaalitodellisuuslasien lisäksi sovelluksen käyttöön tarvitaan jonkinlainen peliohjain. Tähän projektiin valittiin Microsoftin XBOX 360 –peliohjain, joka on sekä muotoilultaan, että toiminnallisuksiltaan yksinkertainen ja toimiva. Kyseinen peliohjain on suosittu tietysti XBOX 360 –konsolin ohjaimena, mutta se on myös vakiinnuttanut paikkansa yleisimpänä ohjaimena tietokoneilla (How To Geek 2014). Tämän lisäksi ohjaimen toi-

mintoja, kuten tärinämoottoreita, on erittäin helppo ohjelmoida Unity-pelimoottorissa.

Sovelluksen on tarkoitus tulla loppujen lopuksi käyttöön Tuorlan observatorion vanhoihin tutkimustiloihin, joihin on alettu vähitellen rakentaa esittelytiloja. Projektin loppuvaiheessa aiottiin hieman vielä kartoittaa, millaisella kokoonpanolla sovellusta pystyisi käyttämään ja kuinka paljon se tulisi Tuorlan observatoriolle maksamaan, mutta sen raportoiminen jätetään tämän työn ulkopuolelle.

Työn alkuvaiheessa Pasi Nurmen antamien tietojen ansiosta 3D-mallien tarkkuutta ja todenmukaisuutta saatiin parannettua NASAn internetsivuilta löytyvän tietokannan avulla. NASAn tietokannasta löytyy 3D-malleja, joiden joukossa oli työssä käytettävät Kansainvälinen Avaruusasema ja Apollo Kuumoduuli. Näistä malleista saatiin rakennettua tähän sovellukseen tarvittavat mallit, jotka vastaavat riittävän tarkasti oikeita versioita näistä rakennelmista.

NASAn sivuilta löytyvää kuva-, video- ja 3D-malli-dataa saa käyttää opetus- ja informaatiotarkoituksessa vapaasti (NASA 2015), joten se sopi tähän työhön hyvin.

## 6.1 Kuukävely

Sovelluksen ensimmäisessä osassa käyttäjä aloittaa avaruuden tutkimisen 70-luvun lopulla päätökseen tulleen Apollo-avaruusohjelman Kuumoduulin läheisyydessä Kuun pinnalla liikkuen. Liikkumisessa hyödynnetään pelimoottorin fysiikkamoottoria siten, että liikkuminen tuntuu Kuun pinnalla hyppimiseltä. Kuun vetovoima on vain noin yksi kuudesosa Maan vetovoimasta (NASA 2015), joten liikkumisen pitää tuntua kevyeltä. Grafiikan puolesta avaruuspukuna käytetään pohjana samaa 3D-mallia kuin avaruuskävelyssäkin, Kuun pinnanmuodoissa käytetään korkeuskarttaa, Apollo Kuumoduuli rakennetaan NASAn tietokannasta löytyvän 3D-mallin perusteella sekä taustalla näkyvä Maapallo on osa Unity Asset Storesta ostettua pakettia.

### 6.1.1 Ohjelmointi

Ohjelmallisesti kuukävely on erittäin yksinkertainen. Peliohjaimen vasempaa sauvaohjainta hyödynnetään astronautin liikkumiseen. Tähän käytetään yhtä funktiota, jota kutsutaan, kun määritellyt ehdot täyttyvät. Ehtoihin kuuluu to-  
tuusarvo muuttuja *ableToJump*, joka on tosi silloin, kun astronautti on edellisen hypyn jälkeen osunut maahan sekä vektori muuttuja, joka määrittelee peliohjaimen vasemman sauvaohjaimen sijainnin. Sijainti määritellään kahdella arvolla: x-akselilla -1:n ja +1:n välillä, sekä y-akselilla -1:n ja +1:n välillä. Arvon ollessa molemmilla akseleilla 0, tarkoittaa se sitä, että sauvaohjain on keskellä, ns. lepoasennossa.

Vasemman sauvaohjaimen antama arvo tallennetaan kaksiulotteiseen vektori muuttujaan *leftStick*, jonka x-arvona käytetään sauvaohjaimen liikettä ylös ja alas sekä y-arvona sauvaohjaimen liikettä vasemmalle ja oikealle. Tämän jälkeen nämä arvot kerrotaan etukäteen määritellyllä kertojalla *jumpAmountVertical*, jota muuttamalla voidaan manipuloida hypyn pituuksia. Hypyn korkeus taas määritellään *leftStick*-vektorin pituudesta ja kerrotaan etukäteen määritellyllä kertojalla *jumpAmountHorizontal*. Hypyn korkeus myös lukitaan kahden arvon väliin, jotta voidaan määritellä minimi- ja maksimi-arvo ei-toivottujen arvojen estämiseksi. Näillä arvoilla hypyn pituus ja korkeus riippuvat siitä mihin suuntaan ja kuinka kovaa peliohjaimen sauvaohjainta painetaan.

Kun vasemman sauvaohjaimen antaman arvo on muunnettu lopulliseen muotoonsa, se tallennetaan vektori muuttujaan *jumpForceDirection*, jossa x- ja z-akselin arvot määrittelevät hypyn suunnan ja y-akselin arvo hypyn korkeuden. Tätä muuttujaa hyödyntäen voidaan astronautin fysiikkamuotoa liikuttaa voimien avulla. Fyysisiä voimia hyödyntäen astronautti hypyistä saadaan todenmukaisia ja fysiikan lakien mukaisia.

Liikkumisen lisäksi astronauttia voidaan ohjaimen avulla kääntää. Astronautin kääntämiseen käytetään peliohjaimen oikeaa sauvaohjainta samaan tapaan kuin vasenta sauvaohjainta. Käännettäessä käytetään vain sauvaohjaimen x-

akselia, sillä astronautin halutaan kääntyvän vain pystyakselinsa ympäri. Liikkumisen tavoin kääntö tapahtuu fysikaalisesti, eli sauvaohjaimen antama arvo kerrotaan aikaisemmin määritellyllä kertojalla *rotationSpeed* ja tästä saatu lopullinen arvo lisätään astronautin fysiikkamuotoon vääntömomenttina y-akselille. Näin astronautti kääntyy sen perusteella, kuinka paljon ja mihin suuntaan sauvaohjainta kulloinkin käännetään.

Kaiken astronauttiin kohdistuvan voiman kompensoimiseksi fysiikkamuodolla on liikettä ja pyörimistä vastustavat voimat, jotka hidastavat astronautin liikettä ja pyörimistä. Tällä helpotetaan liikkumista ja astronautin ohjaamista.

#### Ohjelmakoodi 1. Astronautin ohjaaminen kuukävelyssä

```
private void MoonWalkControls() {
    if (ableToJump && (leftStick.x != 0 || leftStick.y != 0)) {
        Jump();
    }

    playerRigidBody.AddRelativeTorque(new Vector3 (0, rightStick.x, 0)
        * rotationSpeed);
}

private void Jump() {
    jumpForce = new Vector2 (leftStick.y * jumpAmountHorizontal,
        leftStick.x * jumpAmountHorizontal);

    jumpForceUp = Mathf.Clamp(new Vector2(leftStick.y,
        leftStick.x).magnitude * jumpAmountVertical, 1000,
        jumpAmountVertical);

    jumpForceDirection = new Vector3(jumpForce.y, jumpForceUp,
        jumpForce.x);

    playerRigidBody.AddRelativeForce(jumpForceDirection);

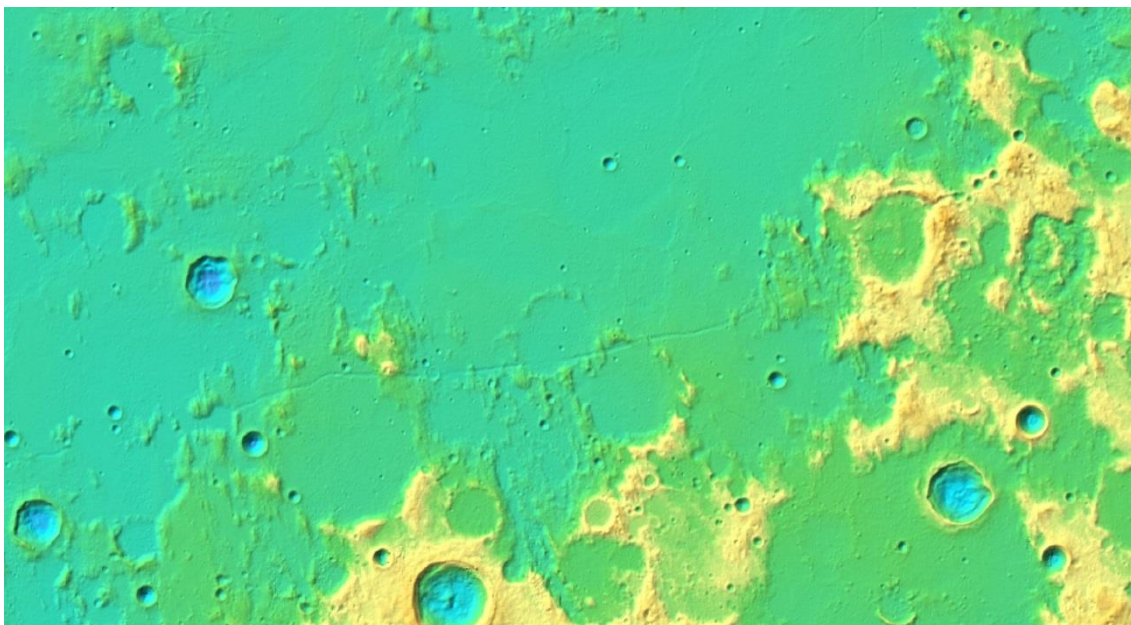
    ableToJump = false;
}
```

### 6.1.2 Grafiikka

Kuukävelyn grafiikka koostuu osittain itse tehdyistä malleista ja tekstuureista sekä internetistä löytyvistä lähteistä, pääosin NASAn arkistoista löytyvistä mal-

leista ja kuvista. Näiden lisäksi käytettiin myös Unity Asset Storesta löytyvää maksullista pakettia, josta löytyy tarkkaan teksturoitu 3D-malli Kuusta, sekä Maapallo, jonka tekstuurit muuttuvat valon mukaan.

Kuun pinnan muodot tehtiin korkeuskartan avulla NASAn LROC:n (Lunar Reconnaissance Orbiter Camera) kuvista. Korkeuskartta saatiin LROC:n internet-sivuilla olevasta QuickMap-palvelusta, jossa pystyy selaamaan paljon erilaista kuvamateriaalia Kuun pinnasta (Arizona State University 2015). Tähän tarkoitukseen käytettiin WAC Color Shaded Relief -kerrosta, joka näyttää Kuun pinnan korkeuden eri väreissä. Tarpeeksi tarkan korkeuskartan saamiseksi lopullinen korkeuskartta tehtiin yhdistelemällä useampi pienempi kuva siten, että lopputulos oli yksi suuri 4097 kertaa 4097 kuvapikseliä oleva kuva. Kuva tallennettiin Unityn Terrain-työkalun vaatimaan RAW-muotoon suoraan Adobe Photoshopista. Korkeuskartan koko seuraa muuten tavallista kahden potenssin mukaista järjestelmää, mutta Unityn Terrain-työkalu vaatii korkeuskartasta luomiseen tekstuurin, joka on  $n^2 + 1 \times m^2 + 1$ , esimerkiksi  $2048 + 1 \times 1024 + 1$  (Unity 2015). Tämä johtuu siitä algoritmista jota pelimoottori käyttää luodessaan 3D-mallia korkeuskartan perusteella.



Kuva 3. Esimerkki korkeuskartasta.

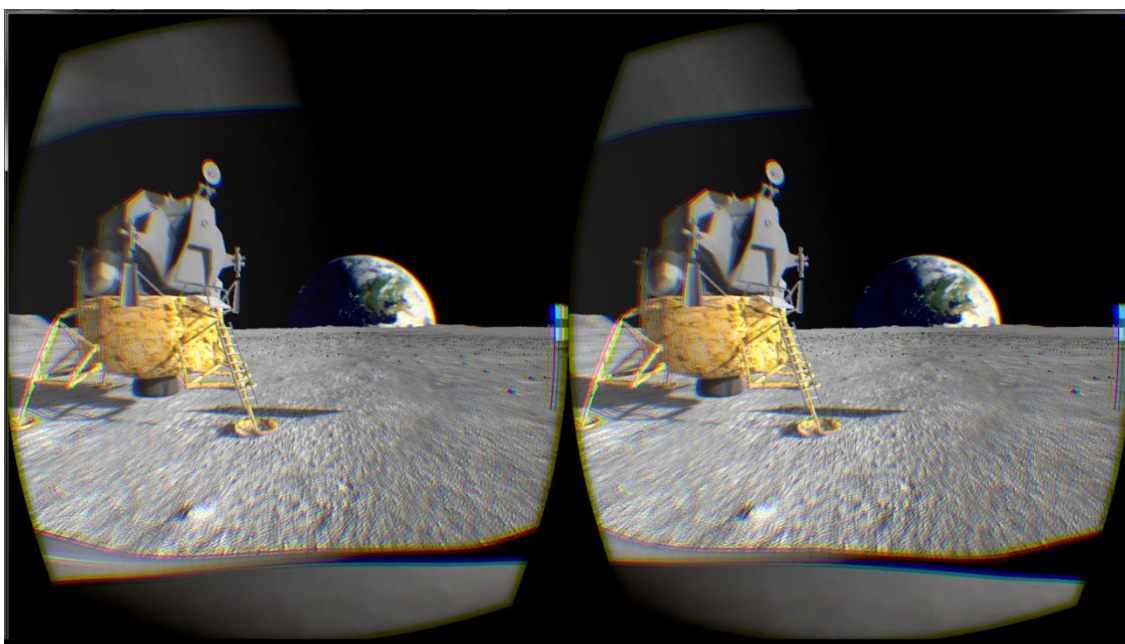
Pinnan tekstuurina käytettiin GalleryHip-sivustolta (GalleryHip 2015) löytynyttä kuvaa Kuun pinnasta. Kuvaa muokattiin Adobe Photoshopilla tarkoitukseen sopivammaksi tekemällä siitä 2048 kertaa 2048 kuvapistettä oleva tekstuuri, joka on toistettavissa monta kertaa peräkkäin ilman, että kahden kuvan välissä näkyy saumaa. Kuvan sisältöä myös piti hieman muokata, että siinä oli vähemmän selkeitä sävyeroja. Suuret värierot tekevät tekstuurin toistosta helposti huomattavamman. Unityn Terrain-työkalu tuki myös niin kutsuttuja normaali-tekstuurikarttoja, joiden avulla 3D-mallien kaksiulotteisille pinnoille voi valon taittumisen ansiosta luoda illuusion suuremmasta geometriadatasta. Normaali-tekstuurikartta luotiin pinnan tekstuurista käyttäen Crazybump-ohjelmaa, jolla pystyy normaalista kuvatekstuurista luomaan yksinkertaisesti normaali-tekstuurikarttoja.

Apollo Kuumoduulin 3D-malli löytyi NASAn arkistosta (NASA 2015). 3D-mallia ei kuitenkaan voinut käyttää sellaisenaan. Tästä johtuen NASAn 3D-mallia käytettiin vain pohjana, jota voitiin käyttää pelikelpoisen 3D-mallin tekemiseen. Tämä helpotti ja nopeutti kuitenkin tekemistä erittäin paljon, sillä edes hyvillä referenssikuvilla on vaikea saada yhtä helposti selkoa monimutkaisista geometrisista pinnoista. Koska kuun pinnalla oli erittäin vähän muuta geometriaa, pyrittiin Kuumoduulista tekemään mahdollisimman yksityiskohtainen. Uudelleenmallintamisen jälkeen 3D-malli teksturoitiin.

Kuun horisontissa näkyvässä Maassa käytettiin samaa mallia kuin avaruuskävelyssäkin, ja se on peräisin Unityn Asset Storesta (Unity 2015). Tämän ostoon päädyttiin monipuolisuuden ja helppouden takia. Paketissa mukana tullut Maapallo sisältää animoidut pilvet varjoineen sekä Auringon valon suunnan mukaan pimeällä puolella näkyvät kaupunkien valot. Maapalloon sisältyy myös valoa taittava ilmakehä, mikä luo realistisen näköisen efektin Auringon mennessä Maapallon taakse.

Astronautin puvun pohjana käytettiin NASAn arkistoista löytyvää 3D-mallia NASAn Extravehicular Mobility Unitista (NASA 2015), eli avaruuskävelyyn tarkoitusta puvusta, jonka selässä on painottomuudessa liikkumiseen tarkoitettu raketireppu. 3D-mallia ei voinut käyttää sellaisenaan pelissä, sillä rakenteeltaan

se oli erittäin rikkonainen ja mallin polygonimäärä oli aivan liian suuri. Uusi 3D-malli tehtiin samaan asentoon ja mahdollisimman paljon alkuperäistä vastaavaksi, mutta siten, että mallin tarkkuus oli peleihin sopivampi. Koska sovellus tapahtuu kokonaan ensimmäisestä persoonasta ja ison osan pelaajan näkökenttää peittää kypärä, ei astronautin puvun ulkonäköön tarvinnut käyttää resursseja. Ulkomuodon piti kuitenkin vastata siluetiltaan mallin alkuperää, jotta pelissä näkyvät varjot näyttäisivät oikeilta. Sen sijaan kypärän sisäosien tarkkuuteen kiinnitettiin enemmän huomiota. Samaa 3D-mallia käytettiin myös avaruuskävelyssä hieman muokattuna. Astronautin puvusta ei myöskään tarvinnut teksturoida kuin kypärän sisäosa, sillä muut osat eivät ole pelaajan nähtävissä. Astronautille tehtiin myös muutama yksinkertainen pieni animaatio hyppyihin ja paikallaan seisomiseen. Näidenkin yksityiskohtaisuus ei tarvinnut olla kovin korkea, sillä pelaaja voi nähdä animaatioita vain omassa varjossaan.



Kuva 4. Kuukävely kehitysvaiheessa.

## 6.2 Avaruuskävely

Avaruuskävely oli alkuperäisessäkin suunnitelmassa, ennen Tuorlan observatorion mukaantuloa, ensimmäinen suunniteltu aktiviteetti. Lopullisessa sovelluk-

sessä avaruuskävely sijoitettiin Kansainvälistä Avaruusasemaa (ISS) ympäröivään avaruuteen. Osasyynä tähän oli se, että ISS on yksi tunnetuimmista ihmisen rakentamasta Maapallon ulkopuolella olevista rakennelmista. Toisena syynä ISS:n valitsemiseen oli se, että NASAn arkistoista löytyi erittäin yksityiskohtainen 3D-malli rakennelmasta. Normaalisti peleissä näin yksityiskohtaisia malleja on vaikea käyttää, mutta koska tässä pelissä on hyvin vähän muita graafisia elementtejä, niin se voitiin käyttää sellaisenaan.

### 6.2.1 Ohjelmointi

Avaruuskävelyn pohjimmaisena ideana oli simuloida rakettirepulla painottomassa tilassa liikkuminen. Liikkuminen toteutettaisiin 6DOF-menetelmällä. Tällä tarkoitetaan kolmea liikkumissuuntaa ja kolmea pyörimissuuntaa. Käytännössä tämä tarkoittaa liikkumista kolmella eri akselilla (x, y, z) ja myös pyörimisliikettä kolmella eri akselilla (roll, pitch, yaw). Liikkumisakselit ovat suoraan kolmiulotteisesta koordinaatistosta. (Road To VR 2013)

Unity-pelimootorissa tämä tarkoittaa käytännössä sitä, että x-akseli on sivuttaissuuntainen liike, y-akseli pystysuuntainen liike ja z-akseli eteen- ja taaksepäin suuntautuva liike. Pyörimisakselit ovat tuttuja lentokoneista, joissa roll tarkoittaa oman akselinsa ympäri pyörimistä. Lentokoneessa tämä olisi keulasta perään menevän akselin ympäri. Pitch on niin sanottu kaltevuusakseli, eli akseli joka menee lentokoneessa siiven kärjestä siiven kärkeen. Tämän akselin ympäri lentokoneen lentokaltevuutta säädellään. Viimeinen pyörimisakseli on yaw, jolla tarkoitetaan koneen läpi pystysuunnassa menevää akselia. Tämä on myös se akseli minkä ympäri esimerkiksi autot kääntyvät.

Aiempien tutkimusten ja kokemusten perusteella pelin ohjaus päätettiin siis tehdä jo aikaisemmin mainitulla Microsoftin XBOX 360 –ohjaimella. Ohjaimesta löytyy liikkumiseen ja kääntymiseen hyvin soveltuvat analogiset sauvaohjaimet. Ongelmana oli kuitenkin se, että sauvaohjaimia on vain kaksi eli ohjaukset vain neljään liikkumisakseliin. Tästä johtuen käyttöön piti ottaa etusormilla käytettävät, ohjaimen etureunassa sijaitsevat lisäpainikkeet. Näistä painikkeista kaksi oli



analogisia liipaisinpainikkeita ja kaksi digitaalisia painikkeita. Päätin sijoittaa yleisemmin tarvittavat ohjaimet, eli liikkumiset eteen- ja taakse sekä sivuille ja pyörimisakselit pitch sekä yaw, sauvaohjaimiin. Nämä siten, että liikkuminen tapahtuu ohjaimen vasemmalla sauvaohjaimella ja pyöriminen oikealla sauvaohjaimella. Näiden lisäksi yli jäävät liikkumisakselit lisättiin siten, että analogisilla liipaisinpainikkeilla ohjattiin astronautin liikkumista ylös ja alas sekä digitaalisilla painikkeilla astronauttia käännettiin viimeisellä pyörimisakselilla, roll.

Avaruuskävelyn liikkumisessa hyödynnettiin Kuukävelystä tuttuja sauvaohjaimen antamia arvoja -1 ja +1 välillä. Sauvaohjaimien arvot otettiin talteen täysin samalla tavalla, kuin Kuukävelyssäkin. Sauvaohjaimien lisäksi käytössä oli myös kaksi analogista liipaisinpainiketta, joista sai molemmilta oman arvon 0 ja +1 välillä. Digitaalisista painikkeista ei saanut mitään numeerista arvoa käyttöön, joten siihen tehtiin yksinkertainen järjestelmä, missä napin painallus antaa muuttujalle *rollLeft* tai *rollRight* arvoksi +1, riippuen kumpaa painiketta painettiin. Painikkeesta irrottaminen palautti muuttujan arvoksi 0. Näitä arvoja käytettiin suoraan ennalta määritellyn kertoimen avulla luomaan joko työntävää voimaa tai kääntävää momenttia astronautin fysiikkamuotoon.

Fyysisen interaktion lisäksi näillä arvoilla säädeltiin peliohjaimesta löytyviä värinämoottoreita. Peliohjaimessa on sisällä kaksi sähkömoottoria, joihin on kiinnitetty epäkesko teräspaino. Moottorin pyöriessä epäkesko teräspaino aiheuttaa ohjaimen värinää. Tätä kutsutaan haptiseksi palautteeksi, jolla tarkoitetaan fyysisen palautteen antamista käyttäjälle värinän avulla (Precision Microdrives 2012). Haptista palautetta käytetään usein peleissä alleviivaamaan ja tehostamaan jotain tiettyä tapahtumaa, kuten räjähdystä tai osumaa. Tämän projektin tapauksessa sitä käytettiin antamaan pelaajalle palautetta siitä, kuinka kovaa astronautin rakettirepun raketit painavat pukua eri suuntiin. Värinää aiheuttavat moottorit sijaitsevan ohjaimen eri päissä, joten koodissa saatiin helposti määritettyä värinät niin, että vasemmalle kohdistuvat liikkeet aiheuttavat värinää oikeanpuoleisessa moottorissa ja oikealle kohdistuvat vasemmassa. Neutraalit liikkeet, kuten eteenpäin kohdistuva työntö, aiheuttaa yhtäläisen värinän molemmissa moottoreissa. Liikkumisen lisäksi astronautille lisättiin kypärän päällä si-

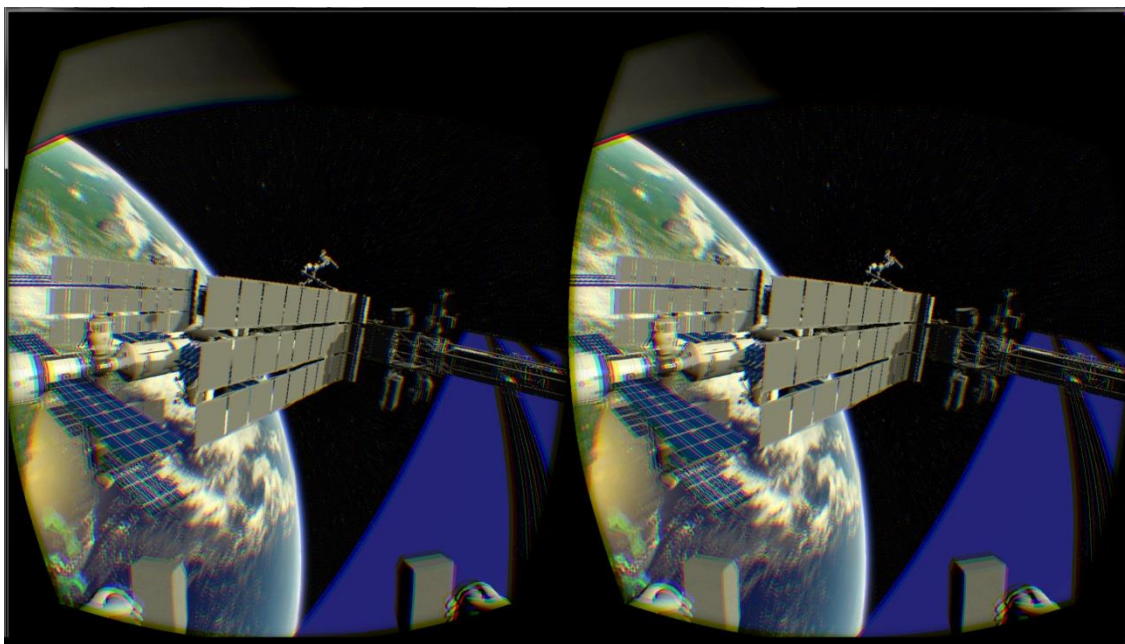
jaitseva kohdevalaisin, jonka voi käynnistää peliohjaimen peukalolla painettavasta A-painikkeesta.

Astronautin lisäksi ohjelmointipuolella avaruuskävelyn lisättiin yksinkertainen skripti, millä voidaan pyörittää pelissä näkyviä objekteja. Tätä käytettiin Maapallon ja Auringon valonlähteenä käytetyn valon kääntämiseen luoden illuusion nopeutetusti Maapalloa kiertävästä Avaruusasemasta.

### 6.2.2 Grafiikka

Avaruuskävelyn huomattavin 3D-malli on taustalla suurena näkyvä Maapallo. Tähän käytettiin samaa Unity Asset Storen valmista, maksullista pakettia, kuin Kuukävelyssä. Tämä nopeutti ja helpotti tätä vaihetta paljon ja auttoi myös saamaan avaruuskävelyn ulkonäön paremmalle tasolle, mitä projektin aikataulun puitteissa olisi muuten saatu tehtyä. Planeetan lisäksi Unity Asset Storen paketista löytyi myös tarkka malli Kuusta, jota tässä myös käytettiin sekä taustana toiminut tähtitaivas.

Avaruusaseman 3D-malli on NASAn arkistosta ja otettu käyttöön sellaisenaan. 3D-malli ei vastaa normaalisti peleissä käytettyjä 3D-malleja, mutta on kuitenkin rakenteeltaan tarpeeksi puhdas. Yksityiskohtia on normaaliin pelikäyttöön liikaa, ja polygonimäärä voisi olla oikeassa peliprojektissa liian suuri, mutta tässä se ei haitannut, sillä pelissä on muuten todella vähän graafisia elementtejä. Pelissä ei myöskään ole tekoälyä, eikä mitään muutakaan raskasta koodipuolen toimintaa, joten suurin osa tietokoneen resursseista voidaan käyttää grafiikan piirtämiseen.



Kuva 5. Kuvankaappaus Kansainvälisestä Avaruusasemasta. Taustalla Maapallo.

Työn loppuvaiheessa jouduttiin tekemään päätös astronauttiin kohdistuvien dynaamisten varjojen poistamisesta Aurinkona toimivasta valolähteestä. Dynaamiset varjot aiheuttivat niin kutsuttua ”z-fightingia”, jossa kaksi täysin yhtä kaukana kamerasta olevaa pintaa näkyy pelaajan kameraan vuorotellen, aiheuttaen pintojen vilkkumisen (Intel 2009). Tämä vaikutti jossain määrin lopulliseen tuotteen ulkonäköön, mutta ei ratkaisevasti. Vastaavasti varjojen aiheuttama pintojen vilkkuminen taas oli niin häiritsevää virtuaalitodellisuuslaseilla nähtynä, että ratkaisu oli helppo tehdä.

### 6.3 Äänet

Alusta asti suunnitelmissa oli luoda äänimaailmasta lähinnä tunnelmaa luova. Tästä syystä sovellukseen ei luotu musiikkeja lainkaan. Pelimaailmassa oli käytännössä kolme ääntä pitävää asiaa: astronautti, astronautin puku ja radiokommunikointi. Astronautin pitämä ääni pyrittiin luomaan vastaamaan ääniä, joita ihminen kuulee ollessaan pienessä suljetussa tilassa. Tätä toteutettaessa äänet suunniteltiin sillä oletuksella, että pelaaja käyttää kuulokkeita. Astronautin pitämä ääni koostuu hitaasta hengityksestä sekä astronautin omasta pulssista.

Taustalla on myös pientä tasaista taustakohinaa, jolla pyrittiin simuloimaan pään sisällä kulkevan veren aiheuttamaa ääntä.

Astronautin itsensä pitämän äänen lisäksi avaruuskävelyssä on äänentuottajana astronautin puku. Puvussa ääntä pitävät pienet rakettimoottorit, joiden äänenvoimakkuus riippuu siitä, kuinka kovin astronautti pukuaan kääntää tai liikuttaa.

Sovelluksen kaikki äänet hain ilmaisten äänien sivustolta Freesound.org. Äänien muokkaamisen ja yhdistelemisen tein Adoben Audition –ohjelmalla.

## **6.4 Kahden sovellusmoduulin yhdistäminen**

Molempien moduulien ollessa valmiit niistä tehtiin yksi kokonaisuus tekemällä sovellukseen päävalikko. Päävalikko koostuu kolmesta painikkeesta, joilla saadaan käynnistettyä sekä molemmat pelimoduulit, että myös sammutettua itse peli. Unity-pelimoottorin helppokäyttöisyyteen kuuluu myös tapa, miten pelien eri kenttiä pystyy vaihtamaan kesken pelin. Jokaisella pelikentällä on järjestysnumero, jonka perusteella kenttää voi ohjelmallisesti vaihtaa haluamaansa.

## **6.5 Oculus Rift – tuen lisääminen ja toiminnan varmistaminen**

Oculus Riftin tuen lisääminen omaan Unitylla tehtyyn pelimoottoriin on tehty erittäin yksinkertaiseksi. Oculus VR:n internet-sivuilta (Oculus VR 2015) löytyy muiden kehitystyökalujen lisäksi tarvittavat kehitystyökalut Oculus Riftin integroimiseen Unity-pelimoottoriin. Integrointiin tarvitaan vain yksi Unity-paketti, mikä sisältää tarvittavat ohjelmakoodit tuen saamiseksi. Työn helpottamiseksi paketissa on myös valmis elementti, mikä sisältää kamerat, ohjelmakoodit ja tarvittavat asetukset, jonka lisääminen omaan peliin ei siis käytännössä vaadi tekijältä mitään säätötoimenpiteitä. Tässä työssä piti hienosäätää tarkemmin ainoastaan kameran sijainti ja astronautin skaala.

Unity-pelimoottori tukee Oculus Riftin Direct HMD Access –käyttömuotoa, joten lopullisessa versiossa virtuaalitodellisuuslaseilla sovellus toimi erittäin hienosti. Minimalistiset valikot joita sovelluksessa tarvittiin, toteutettiin kolmiulotteiseen avaruuteen sijoitettuina, jotta se näyttää stereonäölläkin oikealta.

## **6.6 Käyttökokemukset ja jatkokehitys**

Alkuperäisessä suunnitelmassa oli toteuttaa jonkinlainen käyttäjäkokeilu ja kerätä tämän perusteella palautetta. Tällaista ei kuitenkaan tarvinnut erikseen järjestää, sillä sovellus saatiin yleisölle ensikäyttöön Turun Messuille 21. – 23.8.2015. Käyttäjäkokemukset olivat suurelta osin positiivisia ja suullinen palaute oli hyvää.

Ennen messuja sovellus oli jo käytännössä valmis, mutta siihen tehtiin kuitenkin loppuvaiheessa vielä muutama lisäys. Suurena uutena ominaisuutena tein sovellukseen moninpelin kahdelle pelaajalle. Messuille tämän moninpeli tehtiin vain avaruuskävelylle, sillä avaruuskävely saattaa olla kahdesta moduulista se mielenkiintoisempi vapaalla liikkumisellaan. Uudessa pelimuodossa ei ollut pelaajille mitään uusia toimintoja, mutta siinä pystyy toisen pelaajan kanssa lentämään Kansainvälisen Avaruusaseman ympärillä siten, että pelaajat näkevät toisensa ja halutessaan voivat esimerkiksi törmätä toisiinsa.



Kuva 6. Kuvankaappaus moninpelistä

Moninpelin lisäämisestä johtuen astronautin pukua piti graafisesti päivittää. Koska sovellus oli jo käytännössä valmis ja käytännön osuus hyväksytty, eikä graafinen päivittäminen kuulunut enää opinnäytetyöhön, saatiin graafisen ulkonäön päivittämiseen apua Turku Game Labin henkilökunnalta.

## 7 LOPUKSI

Opinnäytetyön lopputuloksena tehtiin sovellus, joka sisälsi suunnitelmassa olleet kaksi eri. Sovellusta tehdessä pyrittiin ottaa huomioon virtuaalitodellisuuslasien aiheuttamat ongelmat, kuten matkapahoinvointi, mutta sovelluksen ollessa Turun Messuilla esiteltävänä ei kuitenkaan simulaatiopahoinvoinnista annettu palautetta. Tämän sovelluksen tapauksessa siihen saattaa hyvinkin vaikuttaa astronautin 3D-malli, joka on pelaajan nähtävissä lähes koko ajan ja pysyy muuhun maailmaan nähden paikoillaan. Vaikka sovelluksessa on käytännössä koko ajan pelaajalla kiintopisteenä avaruuspuku, joka liikkuu pelaajan mukana, saatavat nopeat liikkeet siltikin aiheuttaa alkuvaiheessa pientä huimausta. Käyttöön tottuu nopeasti ja etenkin avaruuskävelyssä on erittäin helppo liikkua tekemättä suurinopeuksisia liikkeitä.

Graafisesti sovelluksesta tuli riittävän hyvän näköinen. Suurimpana ongelmana oli Kansainvälisen Avaruusaseman suuri koko, mikä teki sen tarkasta teksturoinnista projektin aikataululla käytännössä mahdotonta. Tätä hieman kompensoi kuitenkin se, että 3D-malli on erittäin tarkka. Tästä syystä 3D-malli näyttää hyvältä, vaikka siinä ei tekstuureja olekaan. 3D-mallin tekemiseen ei myöskään tarvinnut käyttää aikaa, sillä se tuli suoraan NASAn arkistoista. Se oli täten myös tarkempi malli avaruusasemasta, mitä olisin itse pystynyt tekemään pelkkien kuvien perusteella, vaikka aikaa olisi ollut useita kuukausia.

Kokonaisuutena sovellus vastaa suunnitelmaa, vaikka sovelluksessa ei tällä hetkellä toiminnallisuuksia liikkumisen lisäksi. Virtuaalitodellisuuslasien julkaisu-aikatauluista johtuen sovellus ei kuitenkaan ole tulossa käyttöön kuin aikaisintaan loppuvuodesta 2015 ja onkin hyvin mahdollista, että sovellusta tullaan vielä kehittämään eteenpäin toiminnallisuuksien ja ehkä myös grafiikoiden puolesta.

Mikäli vastaavanlaista työtä pääsisin tekemään uusiksi, pyrkisin aikatauluttamaan työn niin, että uusien tekniikoiden, kuten pelimoottorien, opettelemiseen voisi varata paljon enemmän aikaa. Vaikka kehitystyökaluna Unity 5 onkin erinomainen, huomaa tietokoneelle kehittäessä, että se on jäänyt teknisesti ja joil-

tain työkaluiltaan kehityksessä jälkeen vertailtaessa suurempiin pelimoottoreihin. Paremmalla aikataululla olisi voitu panostaa tekniseen puoleen enemmän ja käyttää Unity 5:n sijaan esimerkiksi Unreal Engine 4:ää. Sovellusta kehitettäessä vastaan tulleet tekniset ongelmat varsinkin varjojen kanssa olisi saatettu välttää teknisesti kehittyneemmän pelimoottorin kanssa.

Yksi jatkokehityksen kohde voisi olla myös astronautin ohjauksessa. Tästä työssä ei ollut mahdollista hankkia arvokkaita uusia laitteita, vaan ohjaukseen piti käyttää laitetta, joka oli kehitysvaiheessa saatavilla ja tulevaisuudessa mahdollisimman halvalla hankittavissa. Jos peliohjaimeen olisi mahdollista käyttää hieman enemmän rahaa, voisi XBOX 360 –ohjaimen tilalle vaikka hankkia liiketunnistusohjaimet. Näiden avulla voisi myös helpommin kehittää sovellukseen tehtäviä joita pelaajan pitää suorittaa, ja täten suoraan ohjata astronautin käsiä liiketunnistusohjaimilla.



## LÄHTEET

Visbox 2015. Visbox, Inc. Viitattu 5.10.2015. <http://www.visbox.com/products/cave/>

The Rev. Rob Times 2015. Tom Kalinske Talks About His Time Overseeing Sega As Its CEO In the 90s; Reveals That Sega Passed On Virtual Boy Technology, Considered Releasing 3DO. Viitattu 5.10.2015 <http://revrob.com/sci-tech/264-tom-kalinske-talks-about-his-time-overseeing-sega-as-its-ceo-in-the-90s-reveals-that-sega-passed-on-virtual-boy-technology-considered-releasing-3do>

Retro Collect 2015. Nintendo Virtual Boy Celebrates 20th Birthday This Week. Viitattu 5.10.2015. <http://www.retrocollect.com/News/nintendo-virtual-boy-celebrates-20th-birthday-this-week.html>

Hongkiat 2012. Evolution of Mobile Phones. Viitattu 6.10.2015. <http://www.hongkiat.com/blog/evolution-of-mobile-phones/>

Tech Crunch 2014. A Brief History Of Oculus. Viitattu 5.10.2015. <http://techcrunch.com/2014/03/26/a-brief-history-of-oculus/>

Daily Mail 2014. The (virtual) realities of war: Lifelike battlefield simulation teaches medics how to treat wounded soldiers 'under fire'. Viitattu 10.4.2015 <http://www.dailymail.co.uk/sciencetech/article-2656155/The-virtual-realities-war-Life-like-battlefield-simulation-teaches-medics-treat-wounded-soldiers-fire.html>

Medical News Today 2014. Can surgeons be trained using gaming technology. Viitattu 13.4.2015. <http://www.medicalnewstoday.com/articles/281752.php>

Laura L. Arns; Melinda M. Cerney, 2005, The Relationship Between Age and Incidence of Cybersickness Among Immersive Environment Users,

Sixense Entertainment, Inc. 2015. STEM System Full-Body Presence in Virtual Reality. Viitattu 14.4.2015. <http://sixense.com/wireless>

Virtuix 2015. Virtuix Omni. Viitattu 14.4.2015. <http://www.virtuix.com/products/>

Derek Hoiem, University of Illinois 2011. How the Kinect Works. Viitattu 6.10.2015. <http://en.wikipedia.org/wiki/Kinect>

HTC 2015. HTC Vive. Viitattu 14.4.2015 <http://www.htcvr.com/>

A. Fleming Seay; David M. Krum; Laury Hodges; William Ribarsky, 2001, Simulator Sickness and Presence in a High FOV Virtual Environment

Environmental Resources Management Australia, 2009, Parameters of Human Vision and Viewshed Definition

Oculus 2015. Powering The Rift. Viitattu 6.10.2015. <https://www1.oculus.com/blog/powering-the-rift/>

Oculus VR 2015. Development Kit 2. Viitattu 14.4.2015. <https://www.oculus.com/dk2/>

Road To VR 2015. Palmer Luckey and Nate Mitchell Interview: Low Persistence 'Fundamentally Changes the Oculus Rift Experience'. Viitattu 14.4.2015. <http://www.roadtovr.com/ces-2014-oculus-rift-crystal-cove-prototype-palmer-luckey-nate-mitchell-low-persistence-positional-tracking-interview-video/>

Sixense Entertainment, Inc. 2015. STEM System Full-Body Presence in Virtual Reality. Viitattu 15.4.2015. <http://sixense.com/wireless>

Game Career Guide 2008, What is a Game Engine, Viitattu 19.11.2015, [http://www.gamecareerguide.com/features/529/what\\_is\\_a\\_game\\_.php](http://www.gamecareerguide.com/features/529/what_is_a_game_.php)

Digital Tutors 2015. Unity, Source 2, Unreal Engine 4, or CryENGINE - Which Game Engine Should I Choose?. Viitattu 15.4.2015. <http://blog.digitaltutors.com/unity-udk-cryengine-game-engine-choose/>

Mono For Unreal 2015. About. Viitattu 15.4.2015. <https://mono-ue.github.io/about.html>

Unity 2015. Unity. Viitattu 15.4.2015. <http://unity3d.com/unity>

How To Geek 2014. Why You Should Get an Xbox 360 Controller for PC Gaming. Viitattu 5.10.2015. <http://www.howtogeek.com/182825/why-you-should-get-an-xbox-360-controller-for-pc-gaming/>

NASA 2015. Media Usage Guidelines. Viitattu 18.4.2015. [http://www.nasa.gov/audience/formedia/features/MP\\_Photo\\_Guidelines.html](http://www.nasa.gov/audience/formedia/features/MP_Photo_Guidelines.html)

NASA 2015. Moon Fact Sheet. Viitattu 18.4.2015. <http://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html>

Arizona State University 2015. QuickMap. Viitattu 18.4.2015. <http://target.lroc.asu.edu/q3/>

Unity 2015. Unity Documentation – Terrain Other Settings. Viitattu 5.10.2015. <http://docs.unity3d.com/Manual/terrain-OtherSettings.html>

Gallery Hip 2015. Lunar Surface Map. Viitattu 21.4.2015. <http://galleryhip.com/lunar-surface-map.html>

NASA 2015. Apollo Lunar Module. Viitattu 21.4.2015. <http://nasa3d.arc.nasa.gov/detail/lunarlandernofoil-c>

Unity 2015. Unity Asset Store - 8K Earth, Mars and Moon Shaders. Viitattu 21.4.2015. <https://www.assetstore.unity3d.com/en/#!/content/17593>

NASA 2015. Extravehicular Mobility Unit. Viitattu 23.4.2015. <http://nasa3d.arc.nasa.gov/detail/emu>

Road To VR 2015. An Introduction to Positional Tracking and Degrees of Freedom (DOF). Viitattu 5.10.2015. <http://www.roadtovr.com/introduction-positional-tracking-degrees-freedom-dof/>

Precision Microdrives 2012. Introduction to Haptic Feedback. Viitattu 5.10.2015. <http://www.precisionmicrodrives.com/haptics-haptic-feedback-vibration-alerting/haptic-feedback-in-detail/an-introduction-to-haptic-feedback>

Intel 2009. Alternatives to Using Z-bias to Fix Z-fighting Issues. Viitattu 5.10.2015. <https://software.intel.com/en-us/articles/alternatives-to-using-z-bias-to-fix-z-fighting-issues>

Oculus VR 2015. Developer Home. Viitattu 24.4.2015. <https://developer.oculus.com/>